Model Update and Progress

L. Ahrens, K. Brown, A. Luccio, N. Malitsky, V.Ptitsyn, N. Tsoupas, T.Satogata, V. Schoefer, et al.

Outline

❑ New Directions (from Retreat 2005)

- Bringing the RHIC online model to the AGS environment
- Adding a closed orbit to the RHIC online model

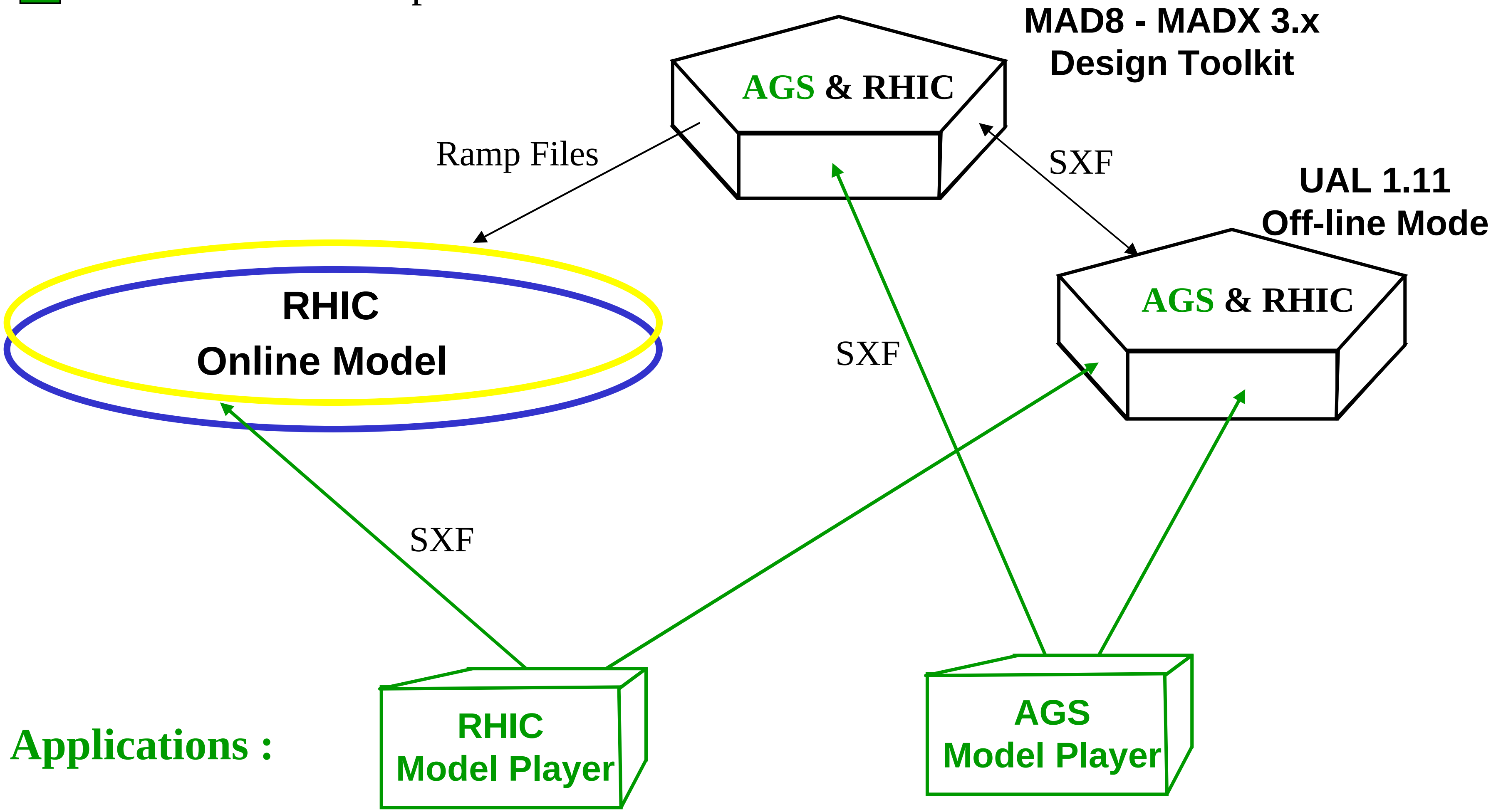
❑ 2005-2006 Development

- Consolidation of the AGS and RHIC off-line modeling environments
- Development and benchmark of the OrbitCalc library for the tune-shift prediction and correction
- Extensions

❑ Next Step

AGS and RHIC Models

- 2005-2006 development



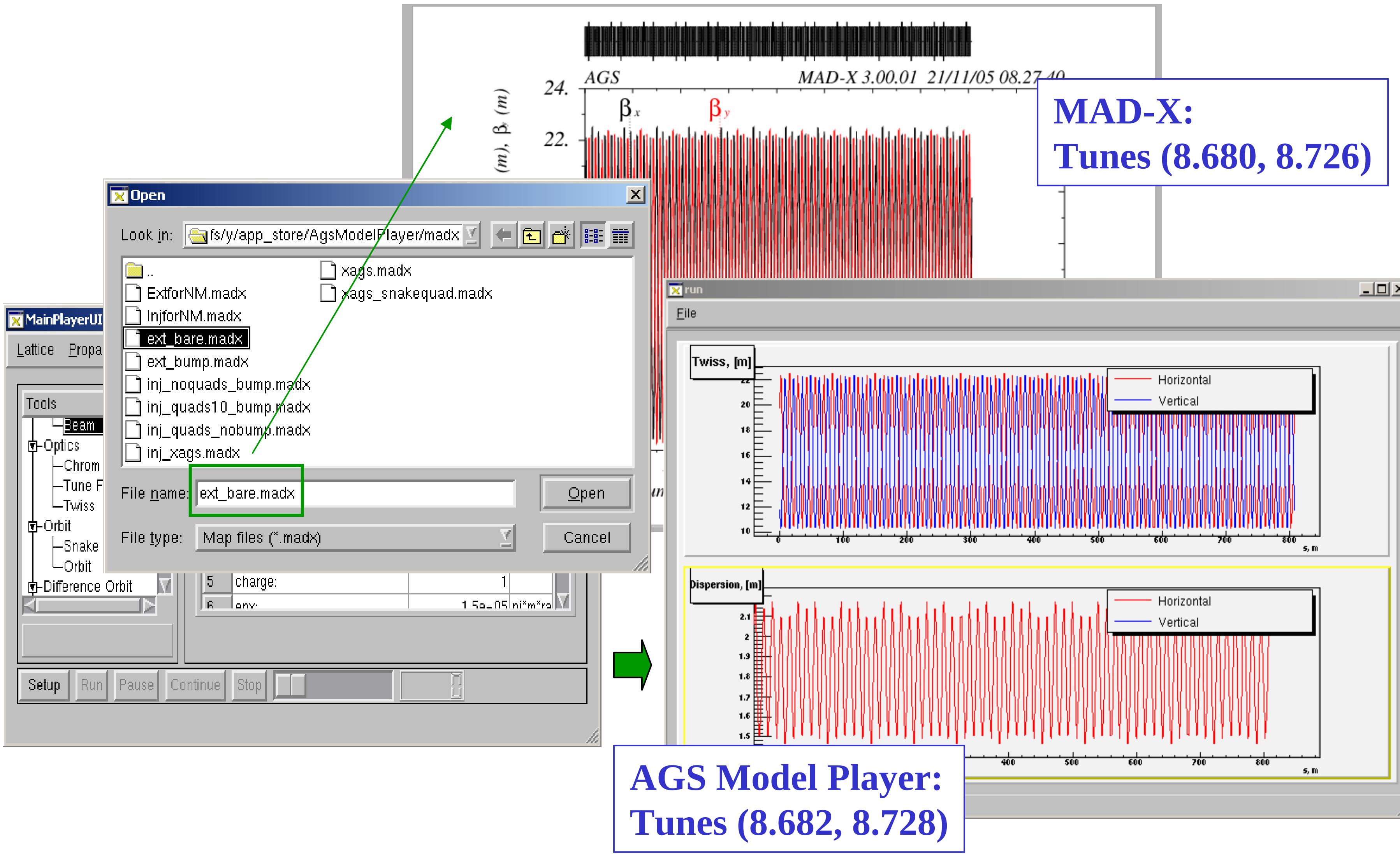
AGS MAD-X Design Model

K.Brown, V. Schoefer, N.Tsoupas

The MAD-X consolidated version has been developed by merging several previous AGS models and divided into the following parts which are available from the /operations/app_store/AgsModelPlayer/madx directory:

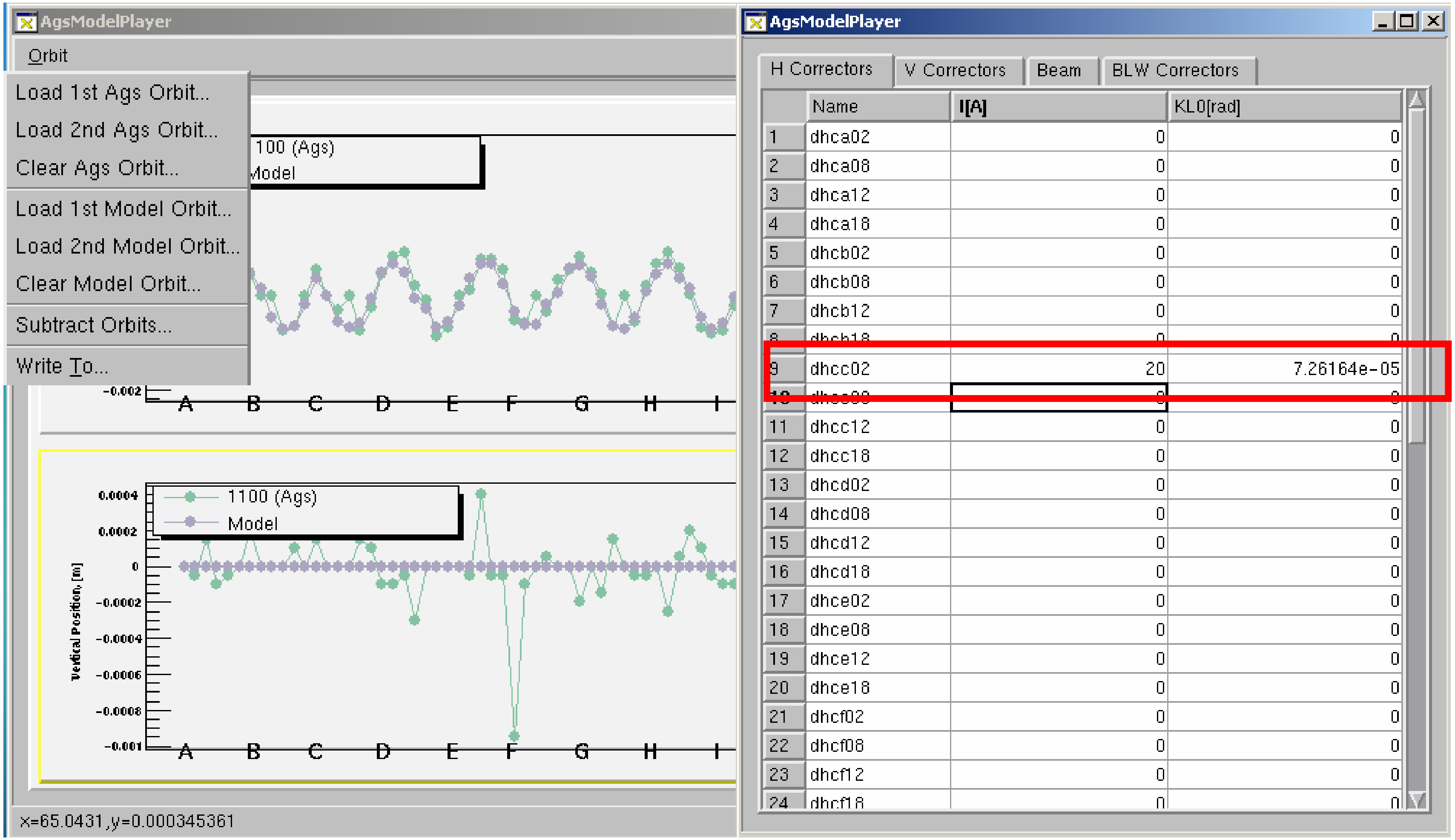
- xags.conversions - polynomial expansions and other expressions required for converting from current to field quantities as required by MAD-X
- xags.rbends_no_k1k2 - AGS main dipoles represented by the MAD-X rbends without K1 and K2 parameters to facilitate the MAD-SXF interface
- xags.lattice - AGS bare lattice
- xags.rbends_add_k1k1 - main dipole K1 and K2 values added with the select comand
- xags.bumps - bumps
- xags.injquads10 - quads compensating the injection optics with 10% snake
- inj_ags.madx - main script with the AGS bare injection lattice. It combines xags.conversions, xags.rbends_no_k1k2, xags.lattice, and xags.rbends_add_k1k2 files
- ext_ags.madx - main script with the AGS bare extraction lattice. It combines xags.conversions, xags.rbends_no_k1k2, xags.lattice, and xags.rbends_add_k1k2 files
- inj_quads10_nobump.madx - main script with the AGS bare injection lattice and compensation quads

AGS MADX-offline model interface



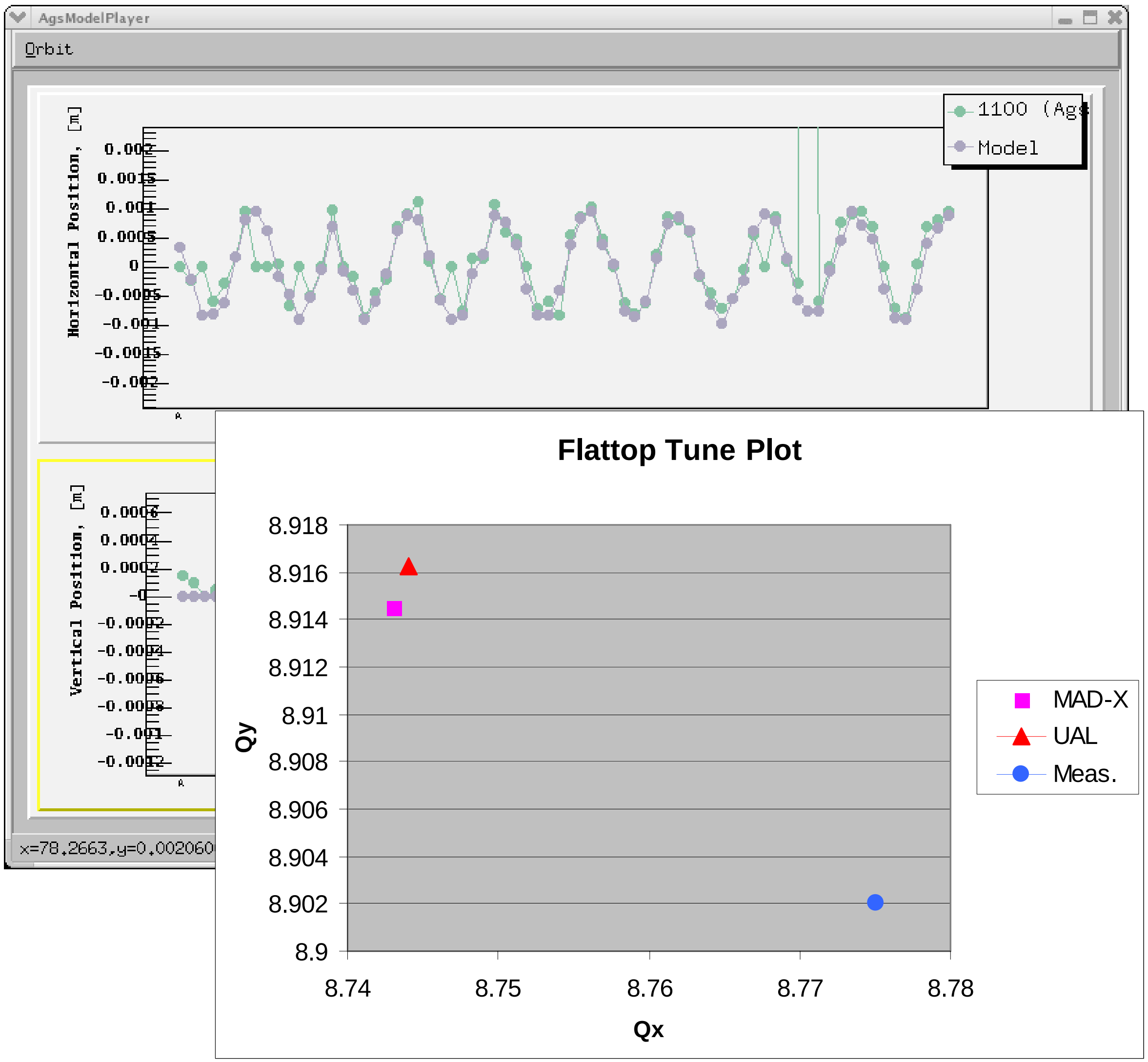
AGS Difference Orbit Display and Control

The intersection point of AGS orbit data and model predictions.



Modeling AGS Flattop

L. Ahrens, M. Bai, K. Brown, S. Vincent, ….



Conditions:

- Live machine settings.
- Dif. orbit with single corrector kick (C08)
- Snakes = drifts.

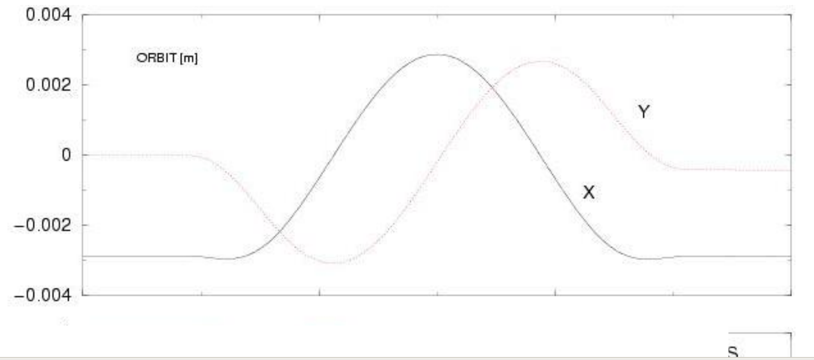
Results:

- Good orbit agreement.
- Large tune difference (model-measured)
  - dQx = -0.03
  - dQy = 0.01
- UAL agrees with MAD-X to ~1.5e-3.

Effects of the cold snake on the injection optics

❑ Orbit offset:

Example of orbit through a partial helical Snake of the AGS type .

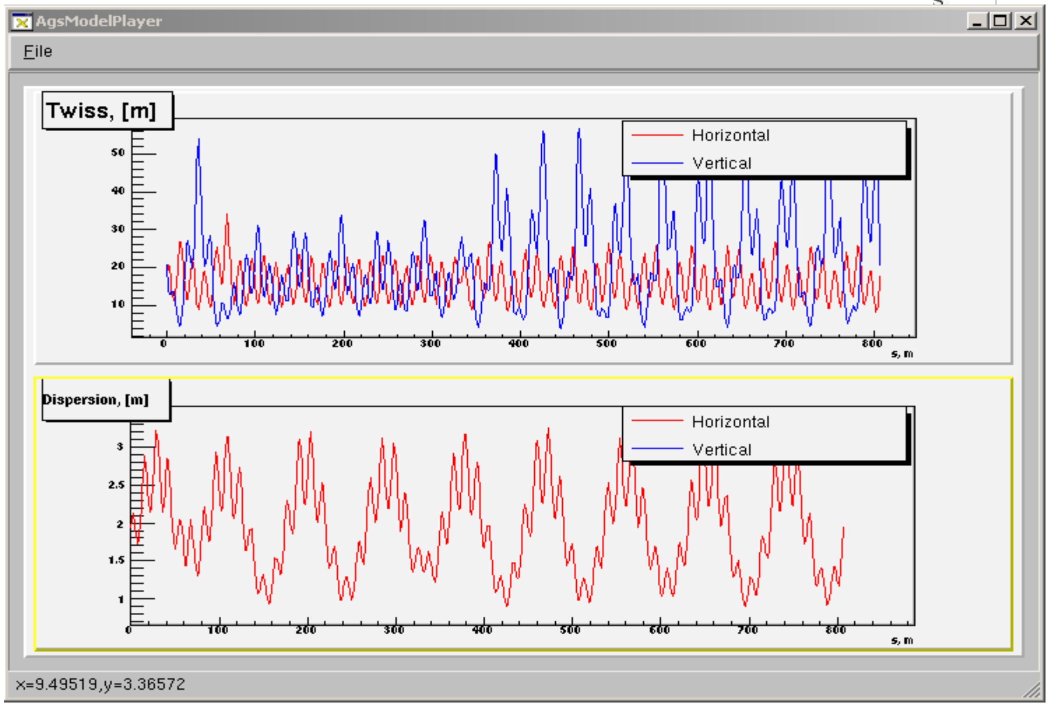


A.Luccio. C-AD/#128 December 2003

❑ Focusing

Lattice functions of the composite injection lattice including:

snake and compensation quads

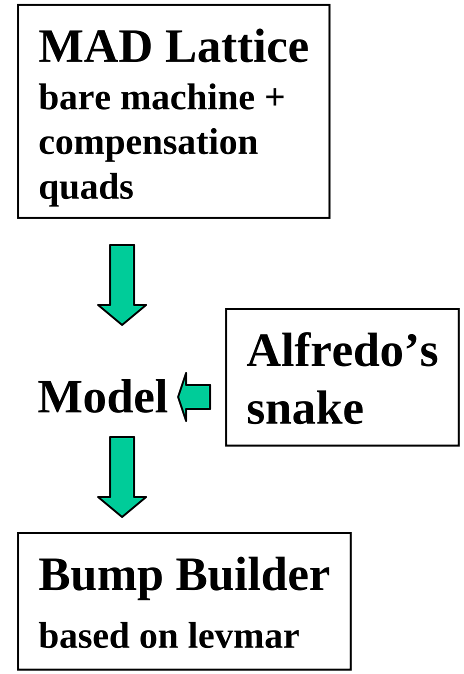


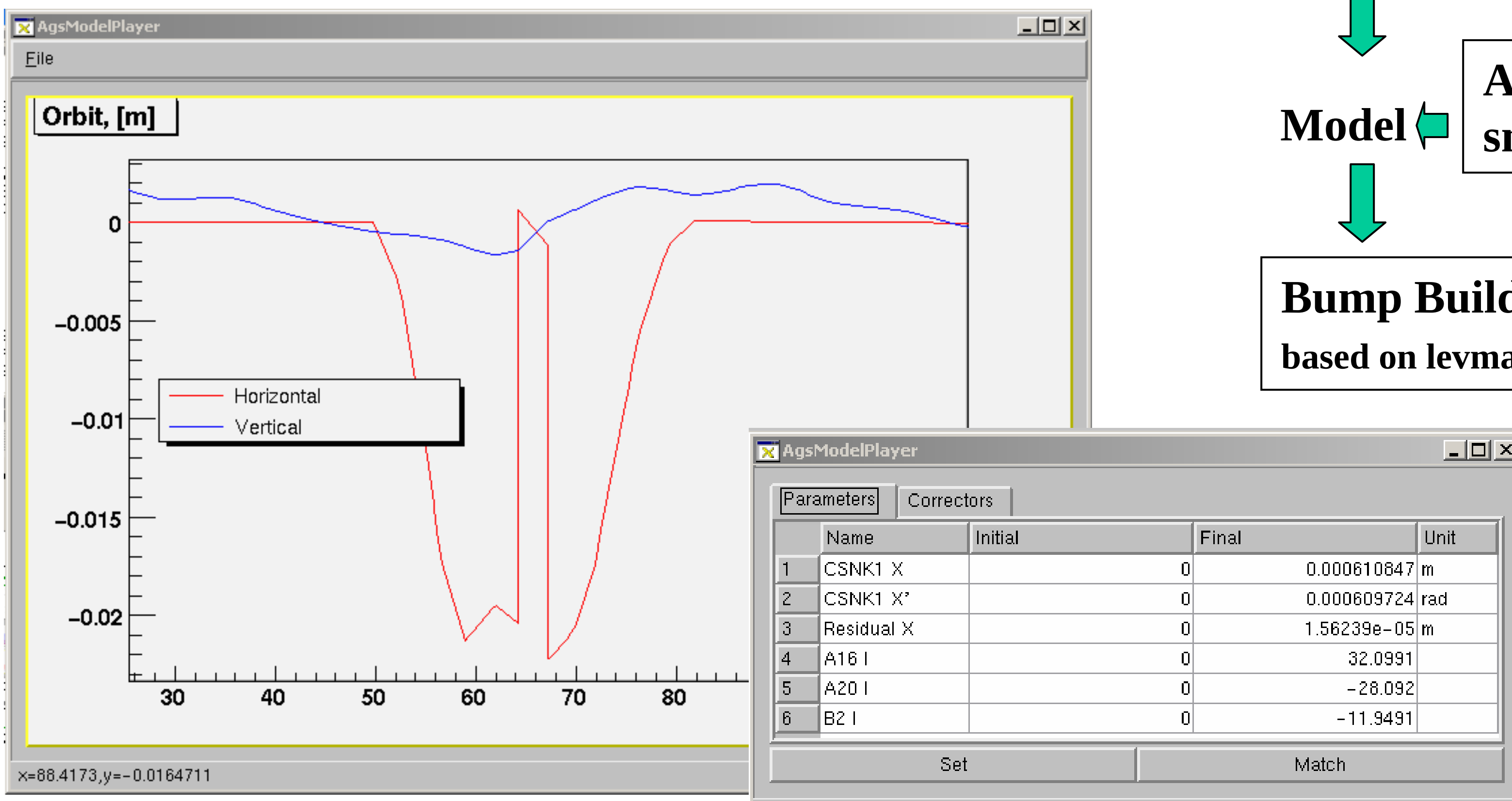
E.Courant, N.Tsoupas

Cold Snake Bump Optimization

N. Tsoupas

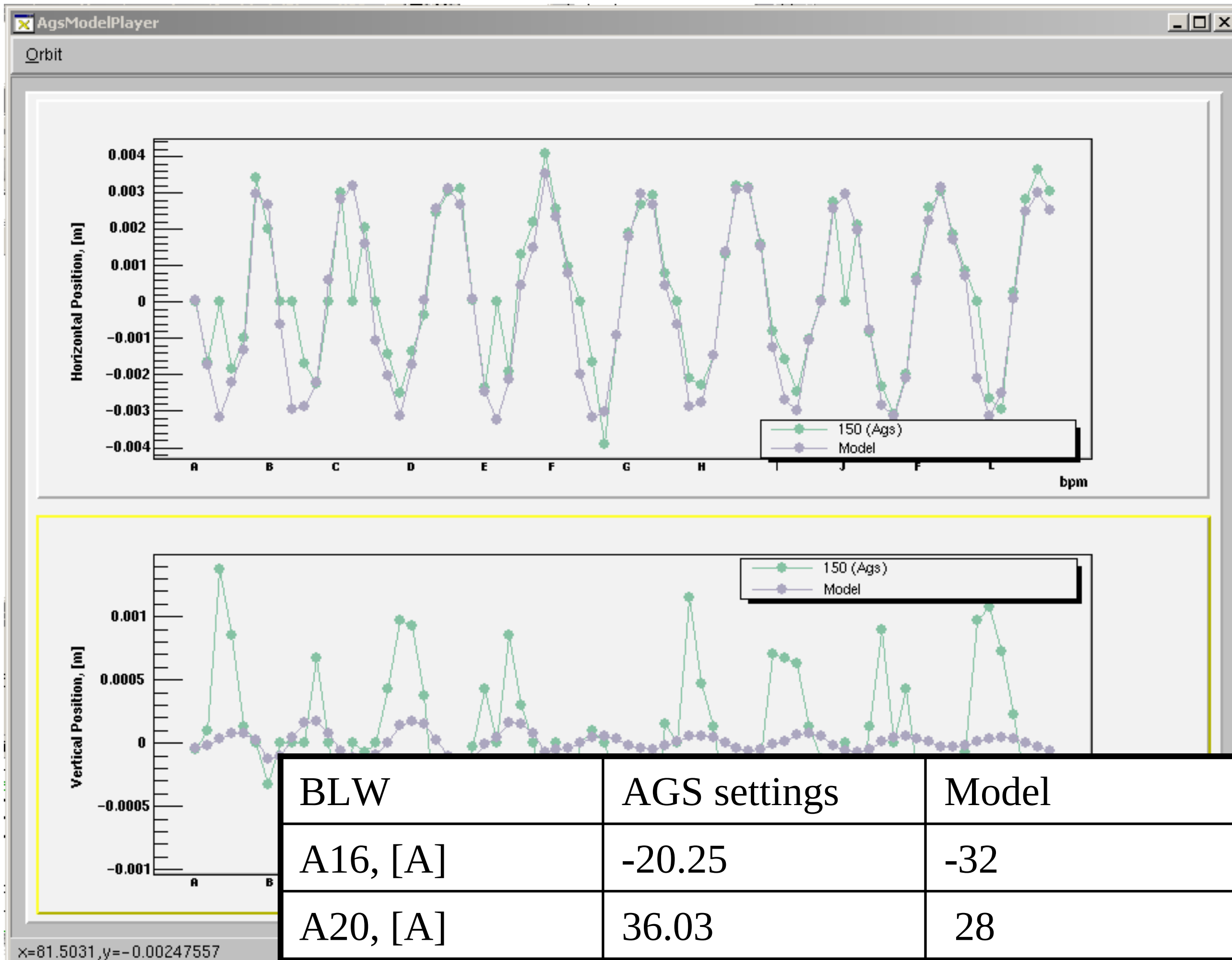
❑ AGS bump optimization is based on N.Tsoupas’ scenario and reuses the levmar library (a C++ version of he MAD lmdif optimization approach) which has already been successfully employed in the RHIC online model environment for β* matching.





	Name	Initial	Final	Unit
1	CSNK1 X	0	0.000610847	m
2	CSNK1 X’	0	0.000609724	rad
3	Residual X	0	1.56239e−05	m
4	A16 I	0	32.0991	
5	A20 I	0	−28.092	
6	B2 I	0	−11.9491	

Injection difference orbit after increasing the snake bump by 10%

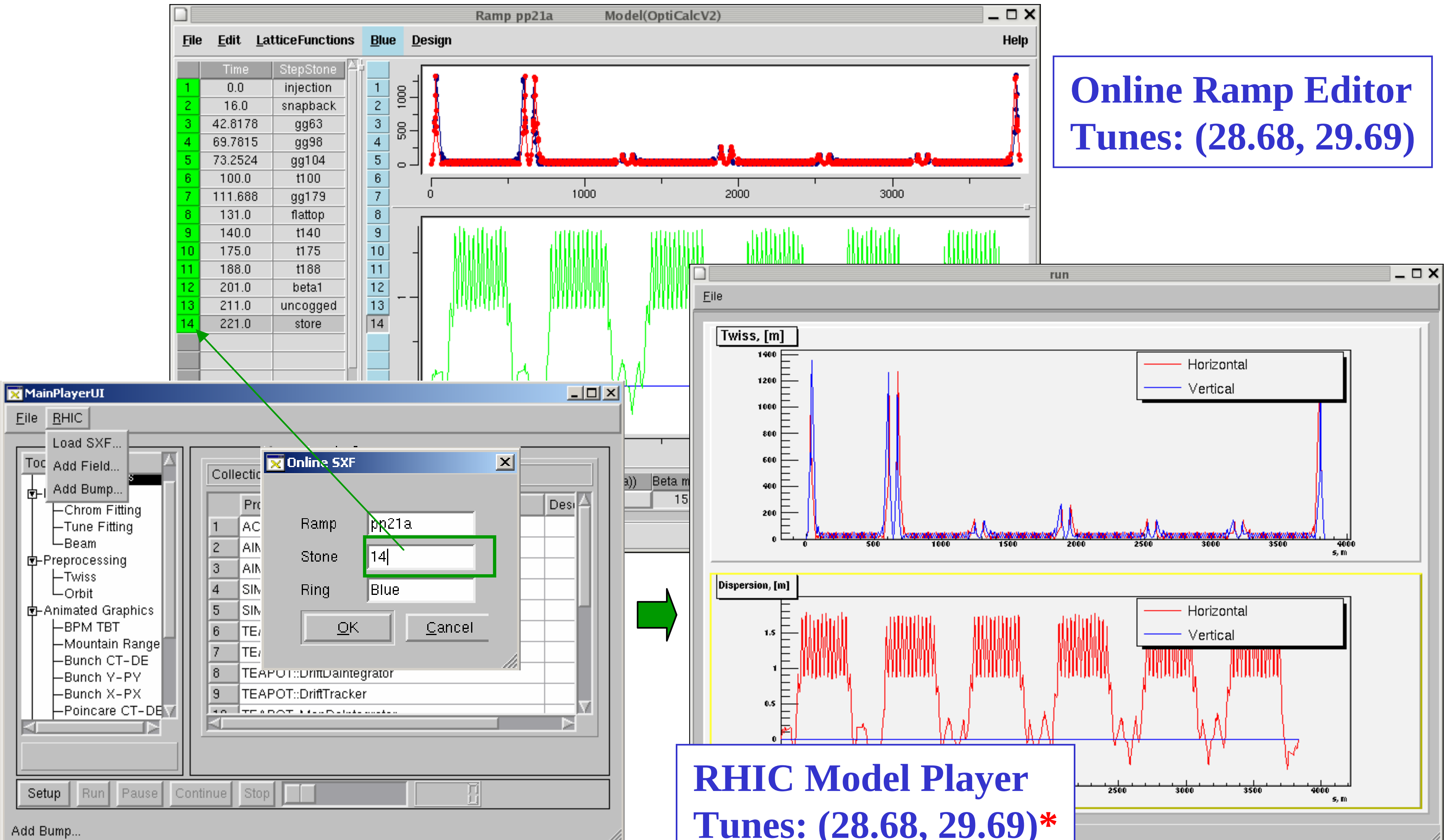


BLW	AGS settings	Model
A16, [A]	-20.25	-32
A20, [A]	36.03	28
B2, [A]	10.96	12

Model:
- Compensation quads = machine settings.
- BLW currents = 10% of machine settings
- Snake = matrix.

Questions:
- Vertical orbit
- Difference between machine and model BLW currents

RHIC online-offline model interface



Online Ramp Editor
Tunes: (28.68, 29.69)

RHIC Model Player
Tunes: (28.68, 29.69)*

* after adding tune shifts from the dipole b2 components

OrbitCalc Library

V. Ptitsyn

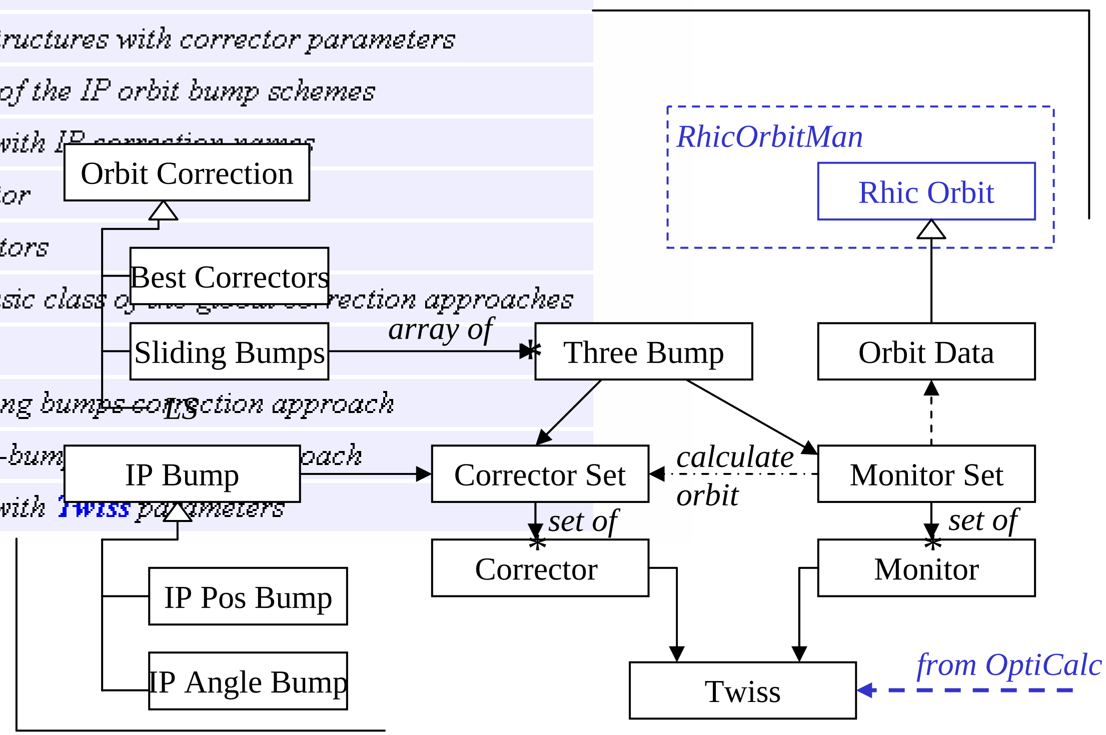
OrbitCalc is a collection of closed orbit algorithms encapsulated from the RhicOrbitDisplay application for reusing and extending by different applications (Sequencer, etc.)

Description :

Here are the classes, structs, unions and interfaces with brief descriptions:

OrbitCalc::Corrector	*Orbit corrector*
OrbitCalc::CorrectorSet	*Set of orbit correctors*
OrbitCalc::CorrStruct	*Vector of structures with corrector parameters*
OrbitCalc::IPBump	*Basic class of the IP orbit bump schemes*
OrbitCalc::IPCorrectionNames	*Container with IP correction names*
OrbitCalc::Monitor	*Orbit monitor*
OrbitCalc::MonitorSet	*Set of monitors*
OrbitCalc::OrbitCorrection	*Abstract basic class of the global correction approaches*
OrbitCalc::OrbitData	*Orbit*
OrbitCalc::SlidingBumps	*Global sliding bumps correction approach*
OrbitCalc::ThreeBump	*Local three-bump correction approach*
OrbitCalc::Twiss	*Container with Twiss parameters*

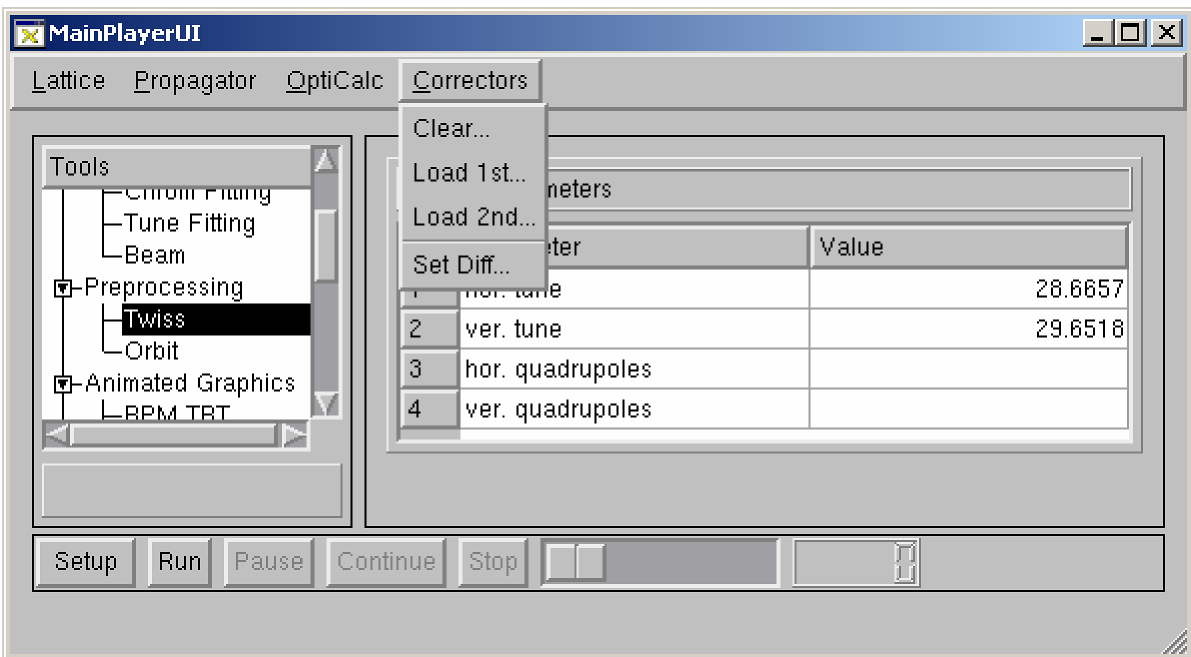
Structure



Calculating the tune shift with the RhicModelPlayer

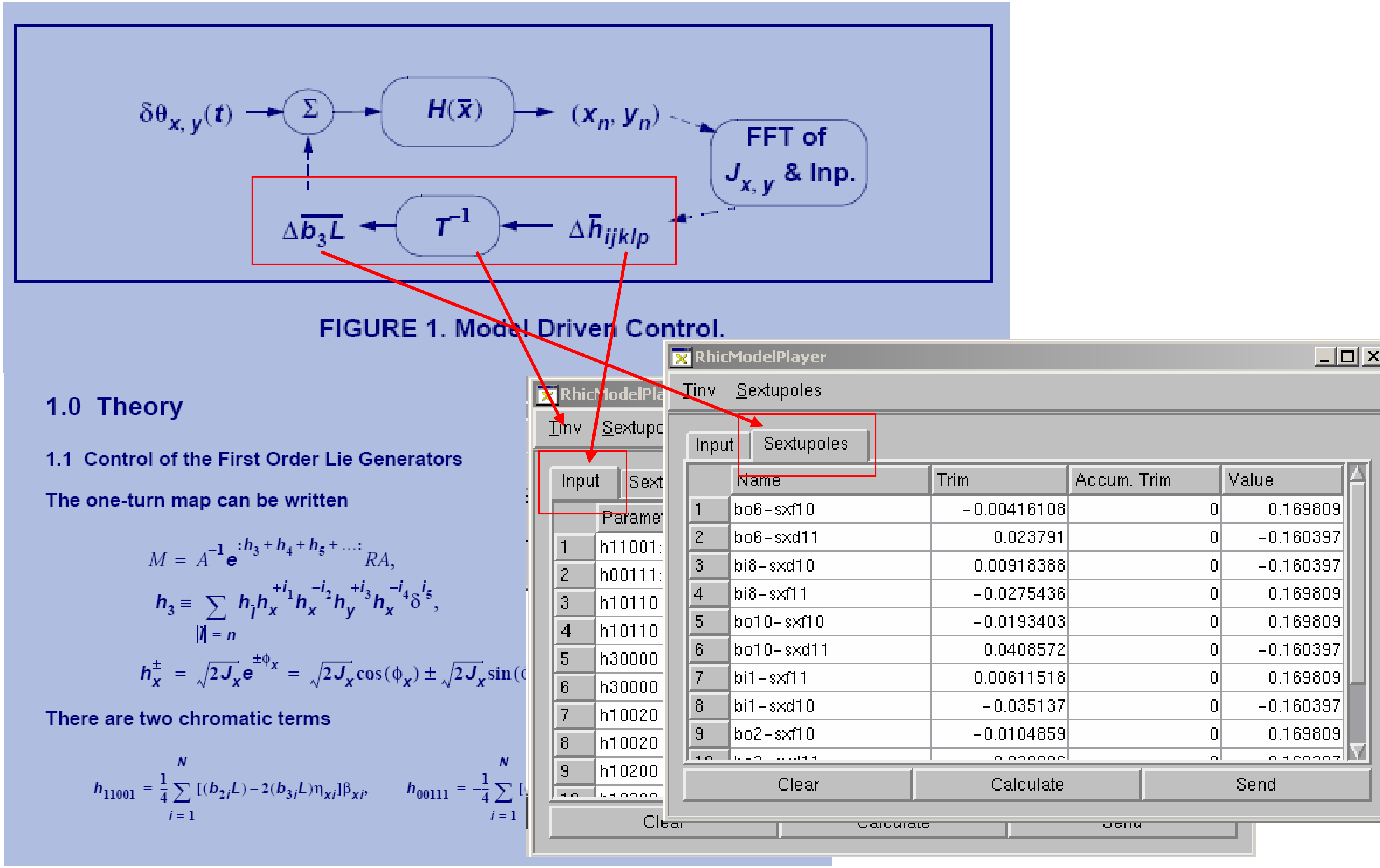
Scenario:

1. **RhicModelPlayer**: loads a model from OptiCalc, loads correctors and calculates tunes (+ Twiss functions) with respect to the closed orbit
2. **RhicOrbitDisplay:** makes a bump
3. **RhicModelPlayer**: loads new correctors, calculate tune shift and compares with the measured data.



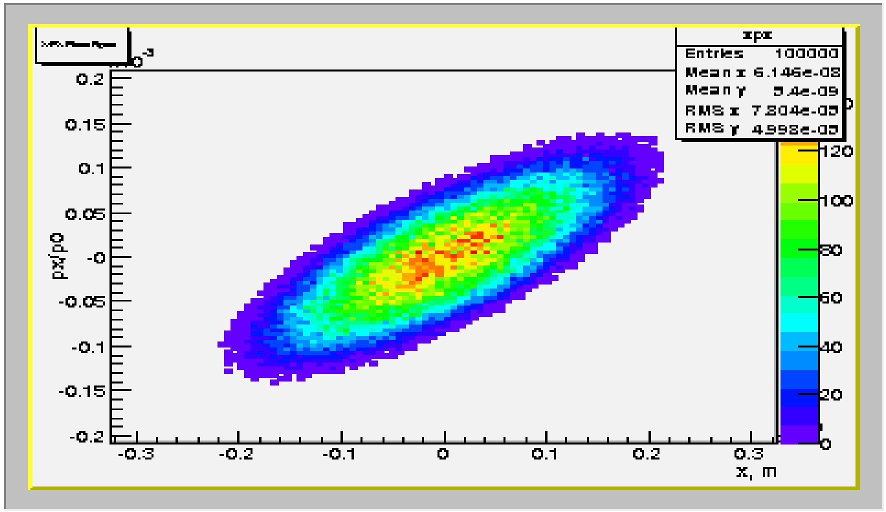
Qx=2/3 correction

J.Bengtsson, Y. Luo

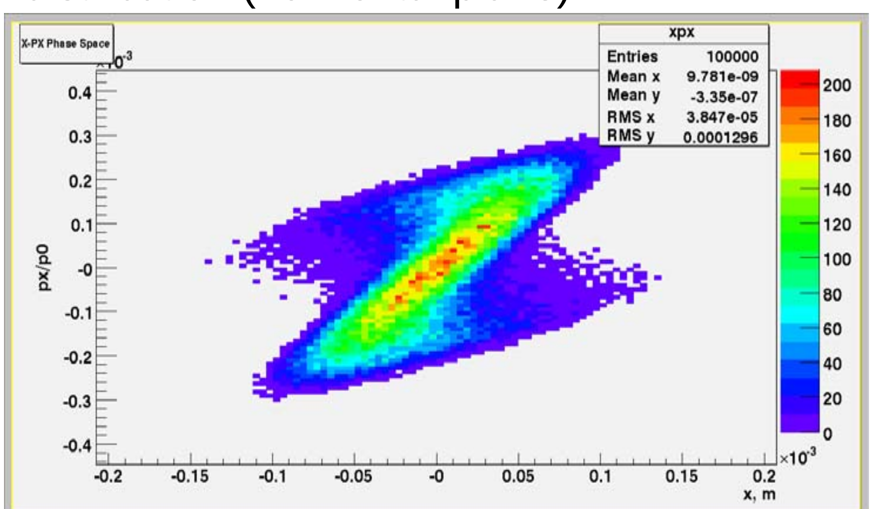


Electron Beam Disruption Simulations in eRHIC

Y. Hao & V.Ptitsyn



Effect of the proton bunch on the electron distribution (horizontal plane)



Beam-Beam Model

$$\begin{pmatrix} E_x \\ E_y \end{pmatrix} = \frac{\lambda(z)}{2\pi\varepsilon_0} \frac{1-\exp[-r^2/(2\sigma_r^2)]}{r^2} \begin{pmatrix} x \\ y \end{pmatrix}$$

- **Proton: Strong Beam**

  transverse plane: Round Gaussian beam
  rms size: 50 microns
  longitudinal plane: Gaussian Beam,
  rms length 20cm,
  sliced

- **Electron: Weak Beam**

  rms size: 50 microns

Next step is to study the electrons' collective effect on protons using strong(electron)–weak(proton) model

Next Step

❑ RHIC OrbitCalc applications

- Merging the RhicOrbitDisplay application with the OrbitCalc library and extensions
- Automating the orbit correction before cogging by extending the RHIC Sequencer with the OrbitCalc-based application.

❑ AGS Extraction model

- Refining the AGS extraction model (gradients, etc) and horizontal and vertical optics based on the orbit response matrix (ORM) measurements.

❑ AGS Injection model

- Reconsidering the cold snake bump optimization based on the analysis of snake models calculated with the different initial orbits.
- Refining the AGS injection vertical optics based on the orbit response matrix (ORM) measurements.

A beam line model for the AtR line

N.Tsoupas

- Finalize the AtR “BPM Application”
  - The measured values of Xcod and Ycod, as derived from the difference of two particular beam trajectories at the location of the BPM’s, will be compared to the corresponding values calculated by the online model of the AtR line.
  - Using the online model of the AtR line, the orbit correction algorithm of the BPM application should be made functional to correct the AtR orbit.
- “Connect” the AtR line with AGS
  - Using the online AGS model, calculate the beam parameters at the beginning of the AtR line. These beam parameters should be compared with the experimentally measured beam parameters using the “Flag application” of the AtR line.